

3.1.4. Business Rules – OpenPKA

STT	Quy tắc nghiệp vụ	Phân hệ / Module	Chức năng liên quan	Mô tả chi tiết	Kết quả kỳ vọng	Công nghệ hỗ trợ
1	CTA dashboard phụ thuộc trạng thái đăng nhập	Trang chính	/ header, hero, CTA	Khi người dùng mở / , hệ thống kiểm tra phiên: đã đăng nhập hiển thị nút tới /hr/dashboard , chưa đăng nhập chỉ cho nút đăng nhập.	Dashboard chỉ mở sau xác thực, tránh lộ dữ liệu.	Next.js App Router, session middleware.
2	Card module điều hướng cố định	Trang chính	Danh sách module chính	Mỗi card ở trang chủ trỏ tới route chuẩn: /org/tree , /hr/employees , /tms , /finance , /documentation . Hover chỉ thay đổi trạng thái UI, không đổi đích.	Người dùng luôn được đưa tới đúng module nghiệp vụ.	Next.js Link, router.push, interactive cards.
3	Org routes bắt buộc đăng nhập trước khi render	Org	/org , route con	Truy cập /org hoặc route con sẽ redirect tới /auth/signin nếu chưa đăng nhập.	Cơ cấu tổ chức chỉ hiện với người đã xác thực.	Middleware Next.js, auth provider.
4	OrgLayout luôn render AppBar + OrgSidebar sau xác thực	Org	OrgLayout	Sau khi pass auth, layout dựng AppBar, OrgSidebar và vùng nội dung làm khung chuẩn cho mọi trang org.	Trải nghiệm thống nhất, tránh thiếu điều hướng.	Layout component, shared UI.
5	Tạo đơn vị yêu cầu nhập code , name , type	Org	/org/unit/create	Form kiểm tra 3 trường bắt buộc; thiếu dữ liệu không cho lưu.	Bản ghi luôn có thông tin nhận diện tối thiểu.	Form validation, Prisma constraints.
6	Đơn vị mới mặc định trạng thái DRAFT và chưa sử dụng	Org	/org/unit/create , danh sách	Sau khi lưu, trạng thái = DRAFT và không được dùng trong báo cáo/phân công đến khi được activate.	Prevent sử dụng đơn vị chưa duyệt.	OrgUnit table, status enum.
7	Hệ thống tự tạo workflow instance cho mỗi đơn vị mới	Org	/org/unit/create	Ngay khi tạo OrgUnit, workflow type ORG_UNIT sinh instance với metadata org_unit_id , code , name .	Quy trình phê duyệt có hồ sơ riêng cho từng đơn vị.	Workflow schema (workflow_instances).
8	Nếu có parent_id , quan hệ cha-con tạo tự động	Org	/org/unit/create	Lưu đơn vị với parent thì sinh OrgUnitRelation loại direct , gán effective_from đúng ngày hiệu lực.	Cây tổ chức luôn khép kín, hạn chế lệch dữ liệu.	org.org_unit_relations , trigger logic.
9	Ghi audit log khi tạo đơn vị	Org	/org/unit/create	Sau khi lưu, audit logger ghi actorId , action: create_org_unit , metadata gồm mã & tên đơn vị.	Có thể truy vết ai tạo và khi nào.	Audit logger, history tables.
10	Ánh xạ trạng thái OrgUnit ↔ action ↔ permission	Org	/org/unit , /org/unit/review	DRAFT cho SUBMIT/RETURN (org_unit.unit.update); REVIEWING cho APPROVE/REJECT (org_unit.unit.approve); APPROVED cho ACTIVATE (org_unit.unit.activate) hoặc RETURN; ACTIVE cho SUSPEND; SUSPENDE/INACTIVE cho ACTIVATE lại.	Mọi hành động phê duyệt tuân thủ phân quyền chuẩn.	Workflow service, RBAC (org_unit.*).
11	Trung tâm phê duyệt Org là kênh duy nhất đổi trạng thái	Org	/org/unit/review	Người quản lý phải lọc và thao tác tại trang review, không đổi trạng thái trực tiếp từ danh sách khác.	Quy trình duyệt tập trung, tránh chính tay.	Review UI, action guards.
12	Quan hệ tổ chức chỉ chỉnh tại /org/unit-relations	Org	/org/unit-relations	Thêm/sửa/xoá quan hệ phải qua màn hình quan hệ; phía khác chỉ xem.	Cây tổ chức được chỉnh có kiểm soát.	Relation management API, dedicated UI.
13	Tất cả route HR yêu cầu đăng nhập	HR	/hr , route con	Người chưa đăng nhập bị chuyển tới /auth/signin trước khi vào module nhân sự.	Bảo mật dữ liệu HR.	Middleware Next.js, auth session.
14	HrLayout dựng AppBar + NewSidebar sau khi xác thực	HR	HrLayout	Layout HR chỉ render sau khi xác thực và luôn gồm AppBar, NewSidebar, vùng nội dung.	Điều hướng nội bộ nhất quán.	Layout component, shared UI.
15	Quản lý nhân sự theo chu trình List → Profile → Edit/Create	HR	/hr/employees , /hr/employees/{id} , /hr/employees/new	Người dùng chọn từ danh sách để vào hồ sơ (tabs). Chỉ trong hồ sơ mới có nút Edit; thêm mới dùng /new .	Hạn chế chỉnh nhầm khi chưa xem hồ sơ.	HR employees UI, tab router segment.

STT	Quy tắc nghiệp vụ	Phân hệ /Module	Chức năng liên quan	Mỗi tab chứa một nhóm dữ liệu; chuyển tab trước khi thao tác để tránh sửa nhầm lĩnh vực khác.	Kết quả kỳ vọng	Công nghệ hỗ trợ
16	Quy tắc nghiệp vụ Ma tab chuyên biệt (Thông tin, Lịch sử, ...)	HR	<code>/hr/employees/{id}</code>			
17	Cây nhân sự luôn hiển thị nhân sự theo nút đã chọn	HR	<code>/hr/org-tree, /hr/org-tree/{id}/employees</code>	Chọn một node thì hệ thống tải danh sách nhân sự thuộc node đó ở route con.	Người dùng thao tác đúng phạm vi đơn vị.	Tree component, filtered queries.
18	Đào tạo, chứng chỉ, học hàm tách thành route riêng	HR	<code>/hr/trainings, /hr/employee-trainings, /hr/qualifications, ...</code>	Mỗi loại hồ sơ có màn hình riêng cho danh sách và chi tiết để tránh trộn dữ liệu.	Hồ sơ phát triển nhân sự rõ ràng.	Specialized pages, filterable grid.
19	Đánh giá & hiệu suất dùng cụm màn hình chuyên biệt	HR	<code>/hr/performance-reviews, /hr/evaluation-periods, /hr/my-evaluations, /hr/evaluation-demo</code>	Các chu kỳ đánh giá, biểu mẫu, kết quả cá nhân được tách route để thao tác đúng vai trò.	Chu trình đánh giá minh bạch và tuần tự.	Evaluation service, review UI.
20	Nghỉ phép & thay đổi nhân sự luôn có lịch sử không xóa	HR	<code>/hr/leave-requests, /hr/leave-requests/history, /hr/employee-changes/history, /hr/employee-logs</code>	Mọi yêu cầu đều ghi nhận lịch sử; UI không cung cấp hành động xóa history.	Có thể kiểm tra lại mọi thay đổi.	History APIs, read-only log views.
21	HR quản lý role, permission, assignment ngay trong module	HR	<code>/hr/roles, /hr/permissions, /hr/role-permissions, /hr/user-roles, /hr/assignments</code>	Việc cấu hình RBAC của HR diễn ra tại module HR để giảm phụ thuộc hệ thống khác.	Quyền nhân sự được kiểm soát tại chỗ.	RBAC service, management forms.
22	HR cung cấp trang đổi mật khẩu nội bộ	HR	<code>/hr/change-password</code>	Người dùng HR đổi mật khẩu thông qua màn hình riêng, không rời module.	Giảm rủi ro chuyển module khi đổi mật khẩu.	Password change API, secure form.
23	HR có cụm báo cáo & hồ sơ cá nhân riêng	HR	<code>/hr/reports, /hr/faculty, /hr/profile</code>	Báo cáo nhân sự, báo cáo khoa và trang hồ sơ cá nhân được gom để người dùng tự tra cứu.	Người dùng tự phục vụ thông tin, giảm tải vận hành.	Reporting pages, profile service.
24	Module TMS yêu cầu đăng nhập trước khi render	TMS	<code>/tms, route con</code>	Người chưa đăng nhập bị trả về <code>/auth/signin</code> , người đã đăng nhập được chuyển tới <code>/tms/dashboard</code> .	Quy trình học thuật an toàn.	Middleware, session guard.
25	TmsLayout dựng AppBar + Sidebar chung	TMS	<code>TmsLayout</code>	Sau xác thực, layout chuẩn với AppBar + Sidebar hiển thị cho mọi route TMS.	Điều hướng nhất quán.	Layout component.
26	Quản lý chương trình đào tạo tách rõ <code>list/detail/structure/edit/create</code>	TMS	<code>/tms/programs, /tms/programs/{id}, /tms/programs/{id}/structure, /tms/programs/{id}/edit, /tms/programs/create</code>	Người dùng phải chọn bản ghi trước khi xem cấu trúc hay chỉnh sửa; tạo mới dùng route riêng.	Không chỉnh cấu trúc nhầm chương trình.	Program service, guarded forms.
27	Luồng học phần tách giữa tạo mới và builder syllabus	TMS	<code>/tms/courses, /tms/courses/create, /tms/courses/build-syllabus, /tms/courses/{id}/syllabus</code>	Duyệt danh sách, tạo mới và xây syllabus diễn ra ở các màn hình khác nhau.	Syllabus luôn gần đúng học phần.	Course builder UI, validation.
28	Quản lý ngành/khóa học có route riêng cho danh sách, tạo, thống kê	TMS	<code>/tms/majors, /tms/majors/create, /tms/cohorts, /tms/cohorts/create, /tms/cohorts/statistics</code>	Người dùng thao tác với ngành và khóa học ở màn riêng, kèm trang thống kê cohort.	Quy trình mở ngành/khóa rõ ràng, dễ báo cáo.	Majors/Cohorts services, statistics page.
29	Trung tâm phê duyệt TMS chỉ bật action khi đủ quyền	TMS	<code>/tms/review</code>	Sau khi lọc tài nguyên và mở modal, hệ thống kiểm tra quyền trước khi hiện nút <code>approve/reject/request_edit/publish</code> .	Ngăn publish ngoài thẩm quyền.	Workflow engine, permission guard.
30	Các màn hình hỗ trợ TMS hiển thị lịch sử, báo cáo, tài liệu cấu hình	TMS	<code>/tms/history, /tms/reports, /tms/documents, /tms/config</code>	Người dùng xem lịch sử, báo cáo, tài liệu và cấu hình ở nhóm màn hình riêng để tham chiếu trước khi duyệt.	Tăng minh bạch khi xử lý học thuật.	Support pages, read-only data views.
31	Finance phải tải danh sách CTĐT trạng thái PUBLISHED trước khi nhập học phí	Finance	<code>/finance</code>	Khi vào module, hệ thống fetch chương trình đào tạo đã publish để người dùng chọn đúng CTĐT + năm học.	Tránh gán giá cho CTĐT chưa ban hành.	Finance layout, curriculum API.
32	Gửi đơn giá tín chỉ phải đi qua API <code>/api/finance/tuition-rates</code>	Finance	<code>POST /api/finance/tuition-rates</code>	Sau khi chọn CTĐT + năm và nhập đơn giá, người dùng submit duy nhất qua API này.	Chuẩn hóa điểm ghi nhận học phí.	Finance API handler, Prisma tuition tables.
				Nếu API trả 409 (đã có dữ liệu) UI	Ghi đề có chủ đích	

33 STT	Xử lý xung đột học phí bằng ủy thác nghiệp vụ	Phân hệ / Module	POST /api/finance/tuition- Chức năng liên quan	Nếu có yêu cầu (sau sơ duyệt), thì yêu cầu người dùng xác nhận ghi đề (Mô tả chi tiết) rồi gửi lại.	Chờ duyệt, kết quả sẽ đẩy lên vòng.	Conflict detection logic, confirm response
34	Sau khi cập nhật học phí phải refresh bảng hiện tại và lịch sử 5 năm	Finance	/finance bảng đơn giá	Thành công → tự động tải lại bảng đơn giá hiện hành và bảng lịch sử 5 năm.	Người dùng thấy kết quả ngay, giảm sai sót.	Client state refresh, history query.
35	Mỗi yêu cầu duyệt (Org/TMS/Finance/HR) bắt buộc sinh workflow instance	Workflow	Các form tạo mới cần phê duyet	Khi người dùng submit yêu cầu cần duyet, hệ thống tạo record trong workflow_instances + approval_records ; đây là nguồn sự thật duy nhất vì chưa bật notification.	Lịch sử duyet tập trung, dễ truy vết.	Workflow schema (workflow_instances , approval_records).